

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	5	(US-6493868-\$ or US-5630131-\$ or US-5652899-\$ or US-5950002-\$ or US-6230314-\$).did.	USPAT	OR	ON	2004/12/22 16:46
S1	4	((("6314558") or ("6397380"))).PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/12/22 11:13
S2	2	((("6314558") or ("6397380"))).PN.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2004/12/22 11:17
S3	266	(717/130).CCLS.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2004/12/22 11:17
S4	16327	(eliminat\$3 remov\$3 delet\$3) near3 probe	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 11:20
S5	4	S3 and S4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 11:18
S6	6	(eliminat\$3 remov\$3 delet\$3) near3 probe near3 redundant	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:24
S7	6	(eliminat\$3 remov\$3 delet\$3) near3 probe with redundant	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:25
S8	37	(eliminat\$3 remov\$3 delet\$3) near3 probe with identical\$2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:48
S9	1253	eliminat\$3 with redundant near2 (info\$7 data)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:27

S10	0	717/130.ccls. and S9	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:27
S11	23	"717".clas. and S9	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:30
S12	0	debug with (function near call) with print\$1 near statement	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:32
S13	1	(function near call) with print\$1 near statement	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:32
S14	1126	(eliminat\$3 remov\$3 delet\$3) with function with entry	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:51
S15	7587	"717".clas.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:50
S16	49	S14 and S15	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 13:50
S17	34	(eliminat\$3 remov\$3 delet\$3) with function with entry with (redundant duplicat\$3 identical\$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:02
S19	0	S17 not entries	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:01
S20	1125	717/106-113.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:02

S21	47904	(eliminat\$3 remov\$3 delet\$3) with (redundant duplicat\$3 identical\$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:02
S22	63	S20 and S21	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:40
S23	18	tail near10 head with (eliminat\$3 remov\$3 delet\$3) with (redundant duplicate identical)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/12/22 15:41



US Patent &amp; Trademark Office

[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

remove redundant node

SEARCH

THE ACM DIGITAL LIBRARY


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)
Terms used **remove redundant node**

Found 21,136 of 147,793

Sort results by

relevance

[Save results to a Binder](#)

Display results

expanded form

[Search Tips](#)☐ Open results in a new window[Try an Advanced Search](#)[Try this search in The ACM Guide](#)

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

Relevance scale ☐ ☐ ☐ ☐ ☐**1 [Removal of redundant dependences in DOACROSS loops with constant dependences](#)**

V. P. Krothapalli, P. Sadayappan

 April 1991 **ACM SIGPLAN Notices , Proceedings of the third ACM SIGPLAN symposium on Principles and practice of parallel programming**, Volume 26 Issue 7

 Full text available: pdf(916.03 KB) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)
**2 [On removing multiple redundancies in combinational circuits](#)**

S.-C. Chang, D. I. Cheng, C.-W. Yeh

 February 1998 **Proceedings of the conference on Design, automation and test in Europe**

 Full text available: pdf(193.67 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)
[Publisher Site](#)

Redundancy removal is an important step in combinational logic optimization. After a redundant wire is removed, other originally redundant wires may become irredundant, and some originally irredundant wires may become redundant. When multiple redundancies exist in a circuit, this creates a problem where we need to decide which redundancy to remove first. In this paper, we present an analysis and a very efficient heuristic to deal with multiple redundancies. We associate with each redundant wire ...

**Keywords:** redundancy removal, implication, logic synthesis, logic optimization**3 [Sequential logic optimization by redundancy addition and removal](#)**

Luis Entrena, Kwang-Ting Cheng

 November 1993 **Proceedings of the 1993 IEEE/ACM international conference on Computer-aided design**

 Full text available: pdf(818.62 KB) Additional Information: [full citation](#), [references](#), [citations](#)
**4 [A new technique for induction variable removal](#)**

Haigeng Wang, Alexandru Nicolau, Roni Potasman

 September 1991 **Proceedings of the 24th annual international symposium on Microarchitecture**

 Full text available: pdf(826.45 KB) Additional Information: [full citation](#), [references](#), [index terms](#)
**5 [Complete removal of redundant expressions](#)**